ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10446723

IHT-INSPIRED NEURAL NETWORK FOR SINGLE-SNAPSHOT DOA ESTIMATION WITH SPARSE LINEAR ARRAYS

Yunqiao Hu and Shunqiao Sun

Department of Electrical and Computer Engineering, The University of Alabama, Tuscaloosa, AL, USA

ABSTRACT

Single-snapshot direction-of-arrival (DOA) estimation using sparse linear arrays (SLAs) has gained significant attention in the field of automotive MIMO radars. This is due to the dynamic nature of automotive settings, where multiple snapshots aren't accessible, and the importance of minimizing hardware costs. Low-rank Hankel matrix completion has been proposed to interpolate the missing elements in SLAs. However, the solvers of matrix completion, such as iterative hard thresholding (IHT), heavily rely on expert knowledge of hyperparameter tuning and lack task-specificity. Besides, IHT involves truncated-singular value decomposition (t-SVD), which has a high computational cost in each iteration. In this paper, we propose an IHT-inspired neural network for single-snapshot DOA estimation with SLAs, termed IHT-Net. We utilize a recurrent neural network structure to parameterize the IHT algorithm. Additionally, we integrate shallow-layer autoencoders to replace t-SVD, reducing computational overhead while generating a novel optimizer through supervised learning. IHT-Net maintains strong interpretability as its network layer operations align with the iterations of the IHT algorithm. The learned optimizer exhibits fast convergence and higher accuracy in the full array signal reconstruction followed by single-snapshot DOA estimation. Numerical results validate the effectiveness of the proposed method.

***Index Terms*—** Sparse linear array, matrix completion, iterative hard thresholding, deep neural networks, single snapshot, direction-of-arrival estimation

I. INTRODUCTION

Millimeter wave (mmWave) radar is highly reliable in various weather environments and antennas can be fit in a small form factor to provide high angular resolution, enhancing the environment perception capabilities. Compared with LiDAR, mmWave radar is a more cost-effective solution, making it crucial for autonomous driving [1]–[3]. Benefiting from multiple-input multiple-output (MIMO) radar technology, mmWave radars can synthesize virtual arrays with large aperture sizes using a small number of transmit and receive antennas [1]. To further reduce the hardware cost, sparse arrays synthesized by MIMO radar technology have been widely adopted in automotive radar [2], [4], [5].

Direction-of-arrival (DOA) estimation is one significant task for automotive radar. Classic subspace-based DOA estimation algorithms such as MUSIC [6] and ESPRIT [7] require multiple snapshots to yield accurate DOA estimates. However, in highly dynamic automotive scenarios, only limited radar snapshots or even just a single snapshot are available for DOA estimation. Consequently, research on single-snapshot DOA methods with sparse arrays is of significant importance. The challenges associated with single-snapshot DOA with sparse arrays are the high sidelobes and the reduction of signal-to-noise ratio (SNR), both of which may cause errors and ambiguity in estimation [4]. If the sparse arrays are designed such that the peak sidelobe level is low [1], compressed sensing algorithms can be utilized to estimate DOA [8], [9]. Alternatively, the missing elements in the sparse arrays can be first interpolated using techniques like matrix completion [2], [4], [10], [11], followed by standard DOA estimation algorithms like MUSIC and ESPRIT. The matrix completion approach exploits the low-rank property of the Hankel matrix formulated by array received signals and completes the missing elements using iterative algorithms [10], [11]. However, typical algorithms for low-rank Hankel matrix completion such as singular value thresholding (SVT) [12] have high computational cost due to the compact singular value decomposition (SVD) in each iteration. In [13], an iterative hard thresholding (IHT) algorithm and its accelerated counterpart fast iterative hard thresholding (FIHT) algorithm were proposed. Both IHT and FIHT feature a simple implementation that utilizes efficient methods for SVD computation and Hankel matrix multiplication during the calculations, and FIHT can converge linearly in specific conditions. However, IHT and FIHT require an appropriate initialization and careful parameter tuning to achieve satisfactory estimates.

Benefiting from the rise of deep learning, deep neural networks (DNNs) with various architectures have been proposed for low-rank matrix completion and show superior performance compared with traditional algorithms [14], [15]. However, these DNNs are usually composed of too many neural layers which leads to a large number of parameters. Furthermore, these DNNs are purely data-driven so they need a huge amount of training data to achieve desirable estimates, which is not available in the scenarios where data collection is expensive.

In this paper, we propose a novel deep learning-based data completion method for sparse array interpolation termed IHT-Net and then apply it for DOA estimation. IHT-Net is constructed following the iteration process in the IHT algorithm but set parameters as learnable. In addition, autoencoder structures are introduced to substitute truncated-SVD (t-SVD) operation in the IHT algorithm. The autoencoders with multiple linear layers can catch low-rank presentations of the signal in the training process, which serves the same purpose as t-SVD. With extensive numerical simulations, we empirically show that the trained IHT-Net outperforms model-based methods such as the FIHT algorithm for both signal reconstruction and DOA estimation using single snapshot.

II. SYSTEM MODEL

A sparse linear array's antenna positions can be considered a subset of a uniform linear array (ULA) antenna positions. Without loss of generality, let the antenna positions of an M-element ULA be $\{kd\}$, $k = 0, 1, \cdots, M-1$, where $d = \frac{\lambda}{2}$ is the element spacing with wavelength λ. Assume there are P uncorrelated far-field target sources in the same range-doppler bin. The impinging signals on the ULA antennas are corrupted by additive white Gaussian Noise with the variance of σ^2. For the single-snapshot case, only the data collected from a single instance in time is available, resulting in the discrete representation of the received signal from a ULA as

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n}, \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_M]^T$, $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2) \ldots, \mathbf{a}(\theta_P)]^T$, with

$$\mathbf{a}(\theta_k) = \left[1, e^{j2\pi \frac{d\sin(\theta_k)}{\lambda}}, \ldots, e^{j2\pi \frac{(M-1)d\sin(\theta_k)}{\lambda}}\right]^T, \tag{2}$$

This work has been funded in part by U.S. National Science Foundation (NSF) under Grants CCF-2153386.

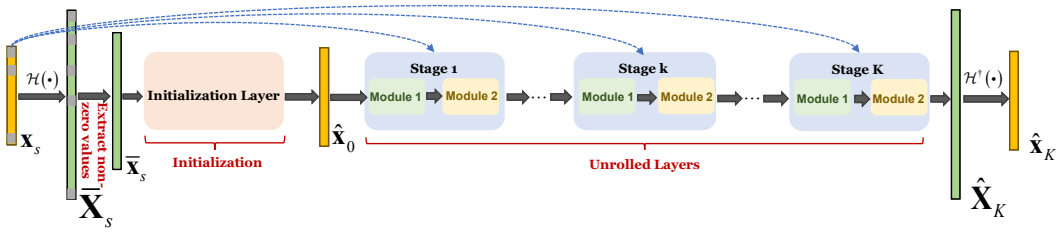


Fig. 1: Illustration of IHT-Net architecture

for $k = 1, \cdots, P$ and $\mathbf{n} = [n_1, n_2, \ldots, n_M]^T$. Then, a Hankel matrix denoted as $\mathcal{H}(\mathbf{x}) \in \mathbb{C}^{N_1 \times N_2}$ where $N_1 + N_2 = M + 1$, can be constructed from $\mathbf{x}$ [16]. The Hankel matrix $\mathcal{H}(\mathbf{x})$ admits a Vandermonde decomposition structure [2], [13], [17], i.e.,

$$\mathcal{H}(\mathbf{x}) = \mathbf{V}_1 \mathbf{\Sigma} \mathbf{V}_2^T, \tag{3}$$

where $\mathbf{V}_1 = [\mathbf{v}_1(\theta_1), \cdots, \mathbf{v}_1(\theta_P)]$, $\mathbf{V}_2 = [\mathbf{v}_2(\theta_1), \cdots, \mathbf{v}_2(\theta_P)]$ with

$$\mathbf{v}_1(\theta_k) = \left[1, e^{j2\pi \frac{d\sin(\theta_k)}{\lambda}}, \cdots, e^{j2\pi \frac{(n_1-1)d\sin(\theta_k)}{\lambda}}\right]^T, \tag{4}$$

$$\mathbf{v}_2(\theta_k) = \left[1, e^{j2\pi \frac{d\sin(\theta_k)}{\lambda}}, \cdots, e^{j2\pi \frac{(n_2-1)d\sin(\theta_k)}{\lambda}}\right]^T, \tag{5}$$

$\mathbf{\Sigma} = \mathrm{diag}([\sigma_1, \sigma_2, \cdots, \sigma_P])$, and $\mathcal{H}(\cdot)$ operator transform signal vector from $M \times 1$ Euclidean space to an $N_1 \times N_2$ Hankel matrix. Assuming that $P \leq \min(N_1, N_2)$, and both $\mathbf{V}_1$ and $\mathbf{V}_2$ are the full rank of matrices, the rank of the Hankel matrix $\mathcal{H}(\mathbf{x})$ is P, thereby indicating that $\mathcal{H}(\mathbf{x})$ has low-rank property [13]. It's worth noting that a good choice for Hankel matrix size is $N_1 \approx N_2$ [18]. Specifically, in this paper, we adopt $N_1 = N_2 = \left(\frac{M+1}{2}\right)$ if M is odd, and $N_1 = N_2 - 1 = \left(\frac{M}{2}\right)$ if M is even.

We utilize a 1D virtual SLA synthesized by MIMO radar techniques [1] with M_t transmit antennas and M_r receive antennas. The SLA has $M_t M_r < M$ elements while retaining the same aperture as ULA. Denote the array element indices of ULA as the complete set $\{1, 2, \cdots, M\}$, the array element indices of SLA can be expressed as a subset $\Omega \subset \{1, 2, \cdots, M\}$. Thus, the signals received by the SLA can be viewed as partial observations of $\mathbf{x}$, and can be expressed as $\mathbf{x}_s = \mathbf{m}_\Omega \odot \mathbf{x}$, where $\mathbf{m}_\Omega = [m_1, m_2, \cdots, m_M]^T$ is a masking vector with $m_j = 1$, if $j \in \Omega$ or $m_j = 0$ if $j \notin \Omega$, and $\odot$ denotes Hadamard product.

Given the aforementioned statements, the Hankel matrix associated with an SLA configuration can be viewed as a subsampled version of $\mathcal{H}(\mathbf{x})$, wherein anti-diagonal entries correspond to the elements of $\mathbf{x}_s$ has values, while the remaining entries are zeros. With the low-rank structure we mentioned before, the missing elements can be recovered by finding the minimum rank of a Hankel matrix that aligns with the known entries [19].

$$\min_{\mathbf{x}} \ \mathrm{rank}(\mathcal{H}(\mathbf{x})) \quad \text{s.t.} \ \ [\mathcal{H}(\mathbf{x})]_{ij} = [\mathcal{H}(\mathbf{x}_S)]_{ij}, \ (i,j) \in \Theta. \tag{6}$$

Here, Θ is the set of indices of observed entries that are determined by the SLA. Note that the rank minimization optimization in (6) is generally an NP-hard problem [19]. In [13] Cai et al. developed iterative hard thresholding (IHT) algorithm for low rank Hankel matrix completion. The convergence speed of IHT algorithm is accelerated by incorporating tangent space projection, resulting in a more expedited variant termed fast iterative hard thresholding (FIHT) algorithm. The main steps in the i-th iteration of the IHT algorithm are as follows

$$\mathbf{X}_i = \mathcal{H}(\mathbf{x}_i + \beta(\mathbf{x}_s - \mathbf{x}_i)), \tag{7}$$

$$\mathbf{x}_{i+1} = \mathcal{H}^{\dagger}(\mathcal{T}_r(\mathbf{X}_i)), \tag{8}$$

where (7) is gradient descent update for current estimate $\mathbf{x}_i$ with fixed step size β on a Riemannian manifold [19]. In step (8), $\mathcal{T}_r$ represents t-SVD for $\mathbf{X}_i$, which projects $\mathbf{X}_i$ onto the fixed-rank manifold to derive a low-rank approximation of $\mathbf{X}_i$. Specifically, it is defined as

$$\mathcal{T}_r(\mathbf{X}_i) = \sum_{k=1}^{r} \sigma_k \mathbf{u}_k \mathbf{v}_k^*, \ \ \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r, \tag{9}$$

where r is the rank of matrix $\mathbf{X}_i$. The operator $\mathcal{H}^{\dagger}(\cdot)$ in (8) is the inverse of $\mathcal{H}(\cdot)$ which maps an $N_1 \times N_2$ Hankel matrix to an $M \times 1$ vector. The IHT algorithm runs iteratively and has fast convergence speed [13]. However, achieving optimal results in IHT requires careful parameter tuning (e.g., step size β and rank r) and can be computationally expensive due to the t-SVD $\mathcal{T}_r$ calculations, particularly in scenarios with large Hankel matrix dimensions.

Once the full array response is obtained, DOA can be estimated using high-resolution DOA estimation algorithms that work for single-snapshot [20], [21].

III. IHT-NET FOR LOW RANK HANKEL MATRIX COMPLETION

To take advantage of the merits of the IHT algorithm and network-based methods, IHT-Net maps the IHT update steps to a deep network architecture that consists of a fixed number of phases, each mirroring one traditional IHT iteration. As shown in Fig. 1, the IHT-Net mainly contains two components: initialization layer, and unrolled layers. The first component provides an initial estimate, analogous to IHT's initialization step, while the second component comprises multiple unrolled layers, mirroring the core iterative steps of the IHT algorithm.

III-A. Initialization Layer

We replace t-SVD in the IHT algorithm with shallow layers autoencoder structures, avoiding the need for rank knowledge and SVD computation. Inspired by the amazing performance of the masked autoencoders [22], we adopt the idea to implement an asymmetric structure that allows an encoder to operate only on observed values (without mask tokens) in the input Hankel vector and a decoder that reconstructs the full signal from the latent representation with mask tokens. As Fig. 1 shows, the gray squares represent mask tokens, with the values set to zeros. Since $\mathbf{x}_s$ is in the complex domain, we concatenate its real part and imaginary part along one dimension and get a $2M \times 1$ vector. This results in the corresponding Hankel vector being twice the original length, with a size of $2N_1N_2 \times 1$, denoted as $\overline{\mathbf{X}}_s$. Next, non-zeros values are extracted from $\overline{\mathbf{X}}_s$, denoted as a new vector $\bar{\mathbf{x}}_s$, along with their positions in the vector denoted as a list ϕ, shared across all layers. The dimension of $\bar{\mathbf{x}}_s$ is $2N \times 1$. As mentioned in [23], inserting multiple extra linear layers in deep neural networks works as implicitly rank minimization of the latent coding. Motivated by this concept, the designed encoder combines 3 linear layers (with bias) separated by rectified linear units (ReLUs). As illustrated in Fig.2(a), the three linear layers share identical input and output dimensions $2N$, aligning with the input vector's length. In our

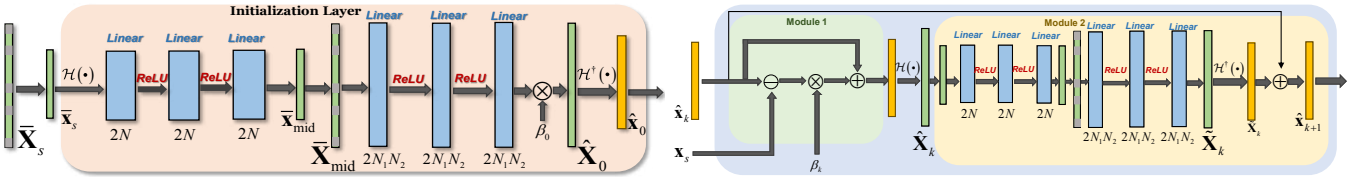


Fig. 2: (a) Illustration of initialization layer of IHT-Net; (b) Illustration of the kth unrolled layer of IHT-Net.

implementation, $2N$ is decided by the number of elements in Θ expressed in (6). We denote the encoder in this layer as $\mathcal{F}_1^{(0)}(\cdot)$, so the output of the encoder is defined as

$$\bar{\mathbf{x}}_{mid} = \mathcal{F}_1^{(0)}(\bar{\mathbf{x}}_s). \tag{10}$$

Then, the output $\bar{\mathbf{x}}_{mid}$ is embedded into a $2N_1N_2 \times 1$ zero vector according to the non-zero values positions in original Hankel vector $\bar{\mathbf{X}}_s$. The vector is denoted as $\bar{\mathbf{X}}_{mid}$ with dimension $2N_1N_2 \times 1$. For the decoder, we follow the same design pattern as the encoder but with input and output sizes set to $2N_1N_2$. Denoting the decoder in this layer as $\mathcal{F}_2^{(0)}(\cdot)$, then the final output of the initialization layer is

$$\hat{\mathbf{X}}_0 = \beta_0 \mathcal{F}_2^{(0)}\left(\mathcal{F}_1^{(0)}(\bar{\mathbf{x}}_s)\right) \tag{11}$$

where β_0 is a learnable scalar, and parameters of $\mathcal{F}_1^{(0)}(\cdot)$, $\mathcal{F}_2^{(0)}(\cdot)$ are all learnable. Finally, the Hankel inverse mapping operates on the output $\hat{\mathbf{X}}_0$ to obtain a $2M \times 1$ signal vector $\hat{\mathbf{x}}_0$, which is expressed as $\hat{\mathbf{x}}_0 = \mathcal{H}^{\dagger}\left(\hat{\mathbf{X}}_0\right)$.

III-B. Unrolled Layers

The k-th unrolled stage consists of two modules. Module 1 is referred to as the Gradient Descent Module, while Module 2 is termed the Low-Rank Approximation Module, which are shown in Fig. 2.

The Gradient Descent Module corresponds to Eq.(7) in the IHT algorithm. With the input $\hat{\mathbf{x}}_k$ from the $(k-1)$-th stage, and $\mathbf{x}_s$ which is broadcasted to every unrolled stage, the intermediate recovery result in k-th stage can be defined as

$$\hat{\mathbf{X}}_k = \mathcal{H}\left(\hat{\mathbf{x}}_k + \beta_k\left(\mathbf{x}_s - \hat{\mathbf{x}}_k\right)\right), \tag{12}$$

where the step size β_k is a learnable parameter.

The Low-Rank Approximation Module keeps the same architecture as the initialization layer while introducing a skip connection within the layers. We firstly extract $\hat{\mathbf{x}}_k$ from the output $\hat{\mathbf{X}}_k$ according to the non-zero values position list ϕ. Then the output of this module is derived by passing it through the encoders-decoders, resulting in

$$\tilde{\mathbf{X}}_k = \mathcal{F}_2^{(k)}\left(\mathcal{F}_1^{(k)}(\hat{\mathbf{x}}_k)\right). \tag{13}$$

After Hankel inverse operation, $\tilde{\mathbf{x}}_k = \mathcal{H}^{\dagger}\left(\tilde{\mathbf{X}}_k\right)$. With the skip connection between the input $\hat{\mathbf{x}}_k$ and the output $\tilde{\mathbf{x}}_k$, the final output of the k-th unrolled stage is

$$\hat{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k + \gamma_k\left(\tilde{\mathbf{x}}_k - \hat{\mathbf{x}}_k\right) \tag{14}$$

where γ_k is a learnable parameter weighting the residual term $(\tilde{\mathbf{x}}_k - \hat{\mathbf{x}}_k)$. The final estimate $\hat{\mathbf{x}}_K$ is obtained after K unrolled stages forward inference.

III-C. IHT-Net Training Specifics

We generate P point-target sources in the same range-Doppler bin. The angles of the sources follow a uniform distribution within the field of view (FoV) spanning $\left[-60^{\circ}, 60^{\circ}\right]$. Their amplitudes have a uniform distribution ranging from $[0.5, 1]$, while their phases are uniformly distributed between $[0, 2\pi]$. Following equation (1), we can generate N_b training labels without noise, denoted as $\{\mathbf{x}_{label}^q\}_{q=1}^{N_b}$ for specific SLA configuration. Then we obtain inputs $\{\mathbf{x}_{input}^q\}_{q=1}^{N_b}$ for network training by adding different levels of Gaussian white noise with SNR randomly chosen from $[10\text{dB}, 30\text{dB}]$ for each training sample. In our experiment $P = 2$ for both training and testing, $N_b = 700000$.

The learnable parameters in k-th phase of IHT-Net are $\left\{\beta_k, \gamma_k, \mathcal{F}_1^{(k)}, \mathcal{F}_2^{(k)}\right\}$. Hence, the learnable parameter set of IHT-Net is $\left\{\beta_k, \gamma_k, \mathcal{F}_1^{(k)}, \mathcal{F}_2^{(k)}\right\}_{k=1}^{K}$. When $k = 0$, the learnable parameters are $\left\{\beta_0, \mathcal{F}_1^{(0)}, \mathcal{F}_2^{(0)}\right\}$.

Given the training data pairs $\{\mathbf{x}_{label}^q, \mathbf{x}_{input}^q\}_{q=1}^{N_b}$ (Note $\mathbf{x}_{label}^q$ and $\mathbf{x}_{input}^q$ are block data which has N training samples pairs), IHT-Net first takes $\mathbf{x}_{input}^q$ as input and generates the output as the reconstruction results denoted as $\hat{\mathbf{x}}_K^q$. We aim to reduce the discrepancy between $\hat{\mathbf{x}}_K^q$ and $\mathbf{x}_{label}^q$, while satisfying the low-rank approximation constraint, which can be stated as $\mathcal{F}_1 \circ \mathcal{F}_2 \approx \mathcal{I}$. Therefore, the loss function for IHT-Net is designed as follows

$$Loss_{total} = Loss_1 + \alpha Loss_2 \tag{15}$$

with

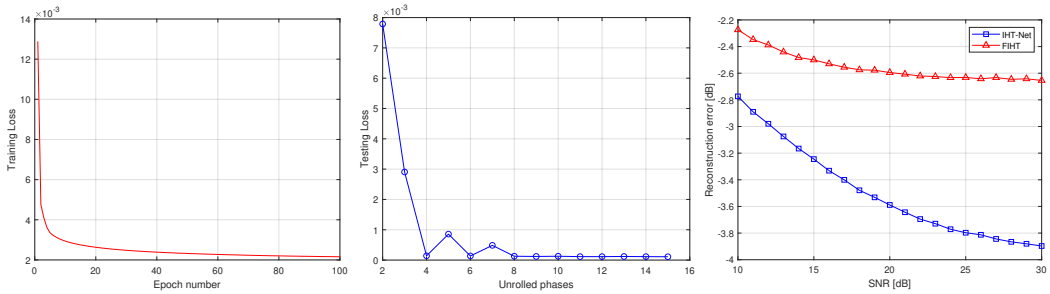
$$Loss_1 = \frac{1}{N_b N}\sum_{q=1}^{N_b}\left\|\hat{\mathbf{x}}_K^q - \mathbf{x}_{label}^q\right\|_2^2, \tag{16}$$

$$Loss_2 = \frac{1}{N_b N}\sum_{q=1}^{N_b}\sum_{k=0}^{K}\left\|\mathcal{H}^{\dagger}\left(\mathcal{F}_1^{(k)}\left(\mathcal{F}_2^{(k)}\left(\hat{\mathbf{x}}_k^q\right)\right)\right) - \hat{\mathbf{x}}_k^q\right\|_2^2, \tag{17}$$

where $K+1$, α are the total number of IHT-Net phases and the regularization parameter, respectively. In our experiments, α is set to 0.01. For training IHT-Net, Adam optimization algorithm [24] is employed with an initial learning rate of 10^{-4}, which decays to 0.5 times of the original rate every 10 epochs.

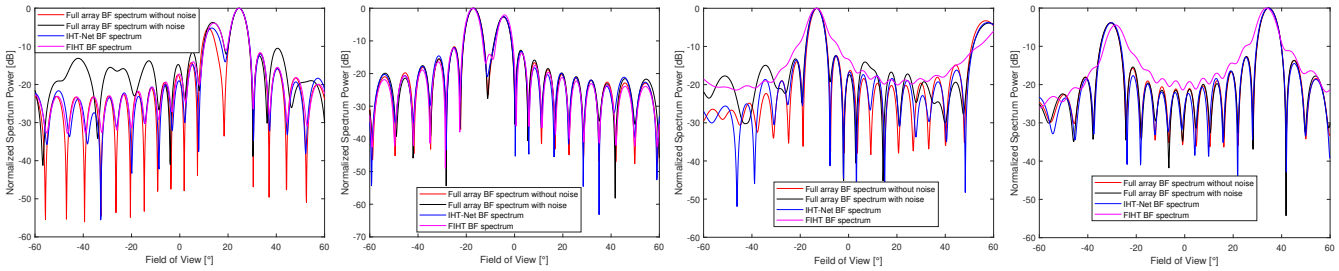
IV. NUMERICAL RESULTS

In this section, we evaluate IHT-Net's performance via numerical simulations. A ULA with $N = 21$ elements is considered, and an SLA is derived from the 21-element ULA by randomly choosing part of its antennas. We first perform experiments using an 18-element SLA, with the training dataset generated following the strategy described in Section III-C. Total 100 epochs of training are conducted in our training procedure. Fig. 3 (a) shows the initial rapid decay of the training loss within the first 10 epochs, which indicates that the proposed IHT-Net is easily trainable. To verify the



(a) (b) (c)

Fig. 3: (a) IHT-Net training loss (defined in (16)) v.s. epoch for IHT-Net with 8 unrolled phases; (b) IHT-Net testing loss (defined in (16)) with various numbers of unrolled phases of IHT-Net; (c) Signal reconstruction error comparison between IHT-Net and FIHT [13] in different SNRs.



(a) (b) (c) (d)

Fig. 4: Beamforming spectrum examples in different SNRs with different SLAs; (a) SNR=10dB, 18-element SLA; (b) SNR=30dB, 18-element SLA; (c) SNR=10dB, 10-element SLA; (d) SNR=30dB, 10-element SLA.

recovery performance of IHT-Net with different layers, we trained IHT-Net with different layers using the same training datasets. Then we randomly generated $5,000$ testing samples in 20dB SNR for evaluation. The testing loss is calculated as (16). Fig. 3 (b) shows that the testing loss decreases as the number of unrolled phases increases, but this decline stabilizes after 8 phases. Thus, we choose to utilize 8 unrolled phases to balance reconstruction performance and computational efficiency in IHT-Net.

Furthermore, we compared IHT-Net and the FIHT algorithm [13] in different SNRs, using a testing dataset of $5,000$ samples per SNR level. In Fig. 3 (c), IHT-Net consistently outperforms FIHT in reconstruction loss, particularly at higher SNR, highlighting its superior performance. We also conducted experiments employing a 10-element SLA, and compared the recovered spectrums in various SNRs with different SLAs. Fig. 4 explicitly shows the beam patterns of recovered full array response by IHT-Net and FIHT, as compared with the spectrums of full array response with and without noise. It can be found that the proposed IHT-Net has denoising ability in relatively low SNR, e.g. 10dB, which indicates that the modules in IHT-Net play the same role as the t-SVD operation in FIHT. In addition, both IHT-Net and FIHT obtain promising spectrums in high SNR e.g. 30dB. Fig. 4(c) and (d) illustrate that for a sparser SLA, FIHT struggles to recover the original signal effectively. In contrast, IHT-Net consistently produces satisfactorily recovered spectrums which keep the mainlobes and sidelobes, confirming its superior recovery performance, particularly with sparser SLAs.

Finally, we compared the mean square errors of DOA estimation using IHT-Net and FIHT reconstruction in different SNRs. We employed beamforming (BF) for DOA estimation. The testing samples number in each SNR is also $5,000$. The errors were calculated using mean-square-loss (MSE). The results shown in

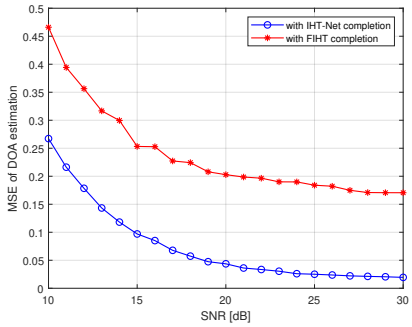


Fig. 5: Comparison of DOA estimation errors after IHT-Net and FIHT [13] completion under different SNRs.

Fig. 5 demonstrated that the IHT-Net completion leads to improved DOA estimation accuracy compared with FIHT completion.

V. CONCLUSIONS

We have demonstrated a novel learning-based sparse array interpolation approach for single-snapshot DOA estimation, termed IHT-Net. It holds potential for applications in automotive radar systems employing sparse arrays. Derived from the FIHT algorithm, IHT-Net incorporates learnable parameters and nonlinear layers, offering an enhanced optimizer through supervised learning with shallow unrolled layers. IHT-Net is easily trainable and interpretable, facilitating further network design and development. Numerical simulations demonstrate its superior reconstruction and DOA estimation performance compared with FIHT.

VI. REFERENCES

[1] S. Sun, A. P. Petropulu, and H. V. Poor, "MIMO radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges," *IEEE Signal Process. Mag.*, vol. 37, no. 4, pp. 98–117, 2020.

[2] S. Sun and Y. D. Zhang, "4D automotive radar sensing for autonomous vehicles: A sparsity-oriented approach," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 4, pp. 879–891, 2021.

[3] M. Markel, *Radar for Fully Autonomous Driving*. Boston, MA: Artech House, 2022.

[4] S. Sun and A. P. Petropulu, "A sparse linear array approach in automotive radars using matrix completion," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8614–8618.

[5] S. Sun, Y. Wen, R. Wu, D. Ren, and J. Li, "Fast forward-backward Hankel matrix completion for automotive radar DOA estimation using sparse linear arrays," in *2023 IEEE Radar Conference (RadarConf23)*. IEEE, 2023, pp. 01–06.

[6] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[7] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.

[8] F. Roos, P. Hügler, L. L. T. Torres, C. Knill, J. Schlichenmaier, C. Vasanelli, N. Appenrodt, J. Dickmann, and C. Waldschmidt, "Compressed sensing based single snapshot DOA estimation for sparse MIMO radar arrays," in *2019 12th German Microwave Conference (GeMiC)*. IEEE, 2019, pp. 75–78.

[9] A. Correas-Serrano and M. A. González-Huici, "Experimental evaluation of compressive sensing for DOA estimation in automotive radar," in *2018 19th International Radar Symposium (IRS)*. IEEE, 2018, pp. 1–10.

[10] S. Zhang, A. Ahmed, Y. D. Zhang, and S. Sun, "DOA estimation exploiting interpolated multi-frequency sparse array," in *2020 IEEE 11th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. IEEE, 2020, pp. 1–5.

[11] ——, "Enhanced DOA estimation exploiting multi-frequency sparse array," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5935–5946, 2021.

[12] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[13] J.-F. Cai, T. Wang, and K. Wei, "Fast and provable algorithms for spectrally sparse signal reconstruction via low-rank hankel matrix completion," *Applied and Computational Harmonic Analysis*, vol. 46, no. 1, pp. 94–121, 2019.

[14] J. Fan and T. Chow, "Deep learning based matrix completion," *Neurocomputing*, vol. 266, pp. 540–549, 2017.

[15] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," *Advances in neural information processing systems*, vol. 30, 2017.

[16] G. Heinig and K. Rost, "Fast algorithms for Toeplitz and Hankel matrices," *Linear Algebra and its Applications*, vol. 435, no. 1, pp. 1–59, 2011.

[17] J. Ying, J. F. Cai, D. Guo, G. Tang, Z. Chen, and X. Qu, "Vandermonde factorization of Hankel matrix for complex exponential signal recoveryâapplication in fast NMR spectroscopy," *IEEE Trans. Signal Process.*, vol. 66, no. 21, pp. 5520–5533, 2018.

[18] Y. Chen and Y. Chi, "Spectral compressed sensing via structured matrix completion," in *International conference on machine learning*. PMLR, 2013, pp. 414–422.

[19] B. Vandereycken, "Low-rank matrix completion by riemannian optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1214–1236, 2013.

[20] E. J. Candès and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *The Annals of Statistics*, vol. 35, no. 6, pp. 2313–2351, 2007.

[21] W. Liao and A. Fannjiang, "MUSIC for single-snapshot spectral estimation: Stability and super-resolution," *Applied and Computational Harmonic Analysis*, vol. 40, no. 1, pp. 33–67, 2016.

[22] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[23] L. Jing, J. Zbontar *et al.*, "Implicit rank-minimizing autoencoder," *Advances in Neural Information Processing Systems*, vol. 33, pp. 14 736–14 746, 2020.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.